

virtual**target** 



Manual de Integração SOAP

versão 1.0 / 2007

www.virtualtarget.com.br

introdução

Este manual é destinado àqueles que possuem conhecimentos sólidos em linguagem de programação para integração SOAP (Simple Object Access Protocol).

A ferramenta Virtual Target dispõe da plataforma SOAP para integração com sistemas externos independentemente do banco de dados ou linguagem de programação utilizada.

sincronização de listas de contatos

VISÃO GERAL

É possível manter uma ou mais listas de contatos atualizadas na ferramenta Virtual Target diariamente sem a necessidade da interação humana.

CONFIGURAÇÃO

A função exercida pela ferramenta Virtual Target é de SOAPClient, portanto, é necessário que seja disponibilizado um serviço SOAPServer para fornecer os dados à ferramenta.

Os parâmetros a seguir são necessários para configuração da sincronização:

1. Endereço do serviço SOAPServer
2. Nome de usuário caso a autenticação HTTP esteja habilitada
3. Senha caso a autenticação HTTP esteja habilitada
4. Realm caso a autenticação HTTP esteja habilitada
5. Método a ser chamado
6. Parâmetro a ser enviado
7. Lista existente na ferramenta Virtual Target a que estes contatos deverão ser sincronização

SINCRONIZAÇÃO

Os contatos fornecidos pelo SOAPServer serão analisados pelo SOAPClient a fim de definir, comparando o campo e-mail, se o contato deve ser adicionado ou atualizado na base de dados da ferramenta Virtual Target.

A ligação entre os dados cadastrais de contatos fornecidos pelo SOAPServer e analisados pelo SOAPClient está nas TAGs de campos adicionais (incluindo @NOME, @EMAIL, @NASCIMENTO e @CELULAR que são estáticos) configurados na ferramenta Virtual Target de forma que não deve-se utilizar o sinal @.

Após a sincronização, um novo arquivo é gerado na área Upload de Listas com todos os contatos sincronizados e outro com e-mails inválidos caso existam, como em um processo comum de upload de listas.

LIMITAÇÕES

A sincronização é realizada apenas uma vez ao dia atendendo as exceções por parte da área técnica responsável pela ferramenta Virtual Target.

Para obter maior performance recomendamos que somente os novos registros e os que foram atualizados nas últimas 24 horas sejam fornecidos.

compartilhando informações com aplicativos externos

VISÃO GERAL

A plataforma SOAP é muito útil para agregar recursos às Intranets, Extranets, Websites e até mesmo sistemas para windows, proporcionando a troca de dados em tempo real. Um bom exemplo é a integração do Virtual Target com um CRM para leitura de histórico de relacionamento.

Para utilizar o webservice é preciso criar um novo usuário no painel de controle, por segurança remova todas as permissões. O login e senha deste usuário deverá ser informado em seu script de integração.

A seguir serão listados os métodos disponíveis no webservice e suas particularidades.

Endereço do webservice: <http://webservices.virtualtarget.com.br/index.php>

WSDL: <http://webservices.virtualtarget.com.br/index.php?wsdl=1>

Método: RetornaListas

Descrição: Retorna dados sobre as listas cadastradas.

Parâmetros: login (string), senha (string)

Retorno: array

Método: RetornaQualidadeListas

Descrição: Retorna quantidade de contatos por status.

Parâmetros: login (string), senha (string)

Retorno: array

Método: RetornaContatosBlackList

Descrição: Retorna relação de contatos adicionados ao black list na data informada.

Parâmetros: login (string), senha (string), dia (int 2), mês (int 2), ano (int 2)

Retorno: array

Método: RetornaHistoricoContato

Descrição: Retorna em ordem cronológica as interações realizadas com o contato informado: envios, clicks, views e opt-out.

Parâmetros: login (string), senha (string), contatoEmail (string)

Retorno: array

Método: AdicionaBlackList

Descrição: Adiciona o contato informado ao black list.

Parâmetros: login (string), senha (string), contatold (int), enviold (int), motivos (array)

Retorno: bool

Detalhes: Caso não tenha o ID do envio forneça o valor 0. Os motivos devem estar no formato a seguir.

\$motivos = array();

\$motivos[] = "O conteúdo é irrelevante";

\$motivos[] = "Não solicitei o recebimento";

Método: RetornaEstatisticasDeEnvio

Descrição: Retorna dados estatísticos do envio informado como: e-mails processados, recebidos, visualizados, clicados, devolvidos e opt-out.

Parâmetros: login (string), senha (string), enviold (int)

Retorno: array

Método: CriaNovoEnvio

Descrição: Cria um novo envio de e-mail marketing, retorna o ID deste envio e quantidade de contatos.

Parâmetros: login (string), senha (string), campanhald (int), remetente_nome (string), remetente_email (string), remetente_replay (string), assunto (string), mensagem (string), datahora_programada (date), notificacao_email (string), notificacao_sms (int), filtro (array)

Retorno: array

Detalhes: A data deve estar no formato dd/mm/yyyy hh:mm. O filtro deve estar no formato a seguir.

\$filtro = array();

\$filtro[] = array("id" => 56, "campo" => "CIDADE", "valores" => array("SALVADOR"));

\$filtro[] = array("id" => 57, "campo" => "CARGO", "valores" => array(""));

\$filtro[] = array("id" => 53, "campo" => "ESTADO", "valores" => array("SP"));

Cada posição do array é um filtro que será interpretado por OU. ID é o ID da lista de contatos. Campo é a tag definida em configurações. Valores são os dados que podem estar contidos no campo definido.

Método: AdicionaAtualizaContato

Descrição: Cria um novo contato ou atualiza caso o email informado já esteja cadastrado.

Parâmetros: login (string), senha (string), email (string), nome (string), nascimento (string), celular (int), listas (array), camposadicionais (array)

Retorno: boolean

Detalhes: A data de nascimento deve estar no formato dd/mm/yyyy. O número de telefone deve estar no formato 551155555555 (DDI+DDD+TELEFONE).

Informar as listas das quais o contato estará associado, como a seguir.

\$listas = array();

\$listas[] = array("listald" => 56);

\$listas[] = array("listald" => 57);

\$listas[] = array("listald" => 53);

Informar os campos adicionais, como a seguir.

\$camposadicionais = array();

\$camposadicionais[] = array("tag" => "empresa", "valor" => "Nome da Empresa");

\$camposadicionais[] = array("tag" => "end", "valor" => "Endereço do contato");

\$camposadicionais[] = array("tag" => "bonus", "valor" => "00,00");

Utilize as TAGs de acordo com a página de configurações no seu painel de controle.

Método: CriaNovaLista

Descrição: Cria nova lista e retorna seu código.

Parâmetros: login (string), senha (string), nome (string), descricao (string)

Retorno: int

Método: CriaNovaCampanha

Descrição: Cria nova campanha e retorna seu código.

Parâmetros: login (string), senha (string), nome (string), descricao (string)

Retorno: int

Método: RetornaEmailsEnvio

Descrição: Retorna emails presentes em um envio realizado. A variavel \$retornaEntregues define se retornara os emails entregues com sucesso ou emails com erro no envio.

Parâmetros: login (string), senha (string), enviold (int), retornaEntregues (boolean)

Retorno: array

Método: RetornaEmailsBlackListEnvio

Descrição: Retorna emails que realizaram o opt-out no envio informado.

Parâmetros: login (string), senha (string), enviold (int)

Retorno: array



www.virid.com.br